



USMC Industry Day #2

Software Reprogrammable Payload (SRP)

System Development Overview

5 May 2009

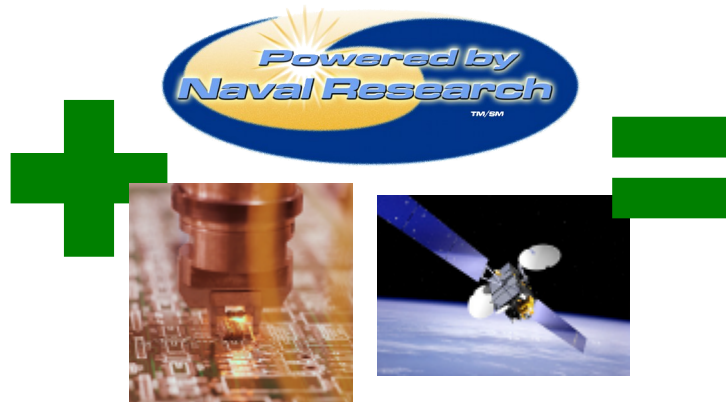
**LtCol Dean Ebert
Aviation Requirements (APW)
HQMC DC/Aviation**

**Christopher Huffine
SRP Team Lead / SE
Naval Research Laboratory Code 8120.1**

How to Rapidly Respond to User Requirements

- Long development and procurement timelines in major acquisition programs
- The world changes a lot in a decade between when the requirements are written and programs are executed
- We think that the solution is a layered approach
 - Provide a framework to support current and future hardware and software
 - Build an acquisition strategy that supports continuous development to be rapidly pushed to users when needed and incrementally otherwise
- Government scientists and engineers working with DoD users to ensure developed standards and requirements will have applicability beyond the short term

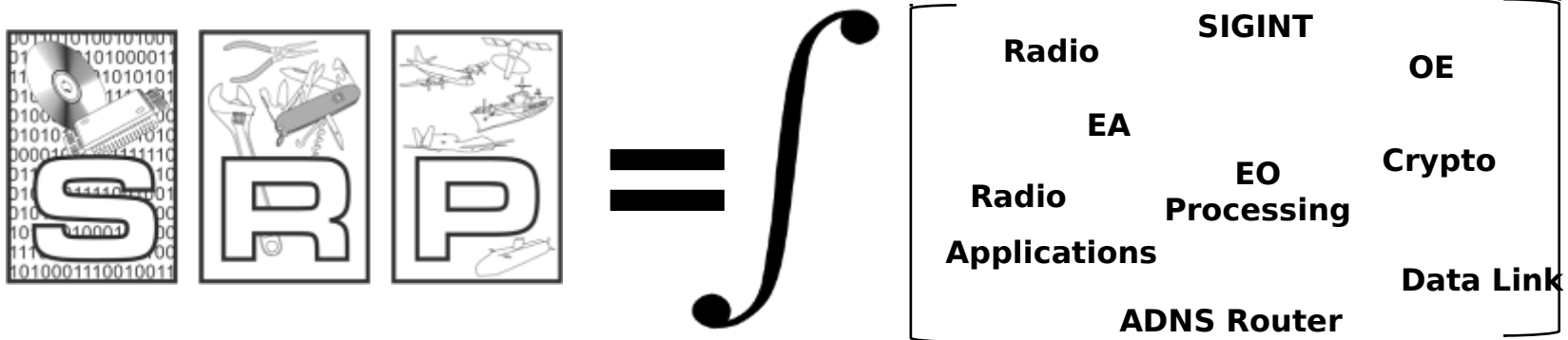
USMC Needs, Requirements and Ideas + Government Labs and Industry =



Mission Success

What is SRP ... Today?

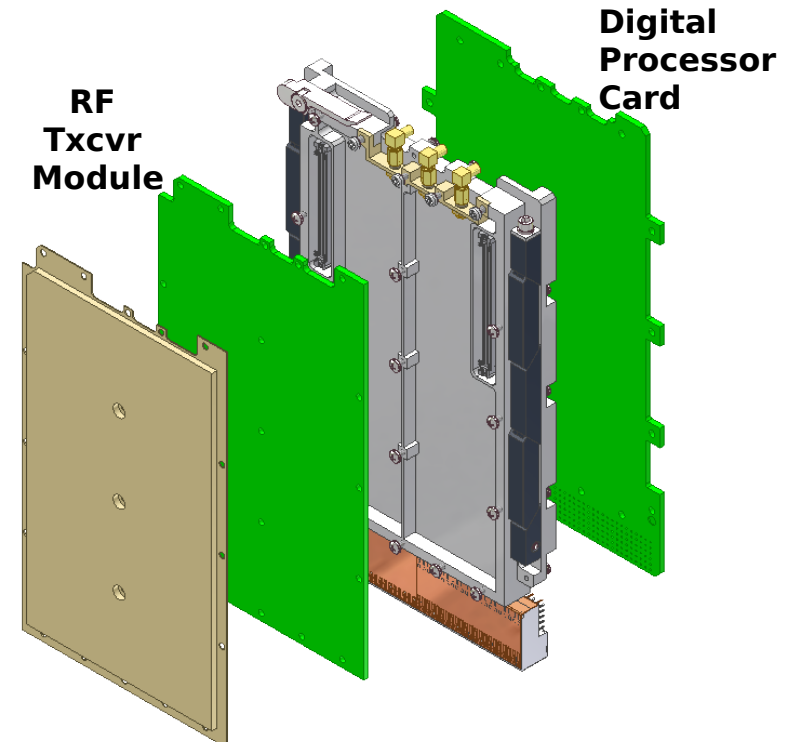
- **Software Reconfigurable Payload**
 - NOT *just* a radio
 - NOT *just* a receiver/SIGINT payload
 - NOT *just* a processor
 - NOT *just* a payload software infrastructure framework
 - NOT *just* a cryptographic module
 - NOT *just* a data or format converter
 - NOT *just* a waveform application



SRP is the result of Integrating A Set of Capabilities Which May Include all or a subset: Comms functions, SIGINT, Crypto, Data Routing, and others not yet conceived of yet

SRP Development Philosophy

- **Government-Industry Team**
 - Involved, committed project sponsors: USMC HQ Aviation, ONR, NAVAIR, OSD RRT0, Navy N6F
 - NRL As Technical Lead
 - A contractor team supplying engineering expertise; not a product
- **Government owned design and software**
 - Government purpose rights at the minimum, open source/GPL whenever appropriate
- **Portable software design**
 - As much industry standard code as possible
 - Minimize specialized software and hardware
- **Modular at all levels**
 - Hardware components
 - Software components
 - Firmware/processing components



Keys to Making SRP Technology Affordable & Effective (1 of 3)

- **Software Reconfigurable Payload technologies hold great promise for allowing diverse use of applications**
- **Keep “proprietary” technology out of SRP applications**
 - **Consider carefully the lifecycle costs of buying into proprietary waveforms and other items vice government development and ownership**
- **Leverage open source (or government source) as much as possible**
 - **Use development tools/toolchains that are “lowest common denominator”**
 - **Manage classification of code and modules carefully**
 - **Share/post source code to be used by others and/or enhanced by others**






Keys to Making SRP Technology Affordable, Effective (2 of 3)






- **Utilize collaboration methodologies**
 - “software forge.mil” being one example in government use - which itself is based on the “Sourceforge” open-source model
- **Government being intimately involved - controllers of the ICDs, and owners of the software products**
- **GFE Software Development Kits and/or Software Development Stations**
 - Level the playing field so SRP application development does not require a million dollar investment in infrastructure
 - Provide access to actual “deployable” hardware suites and test signal generators

SoftwareForge.mil - DoD's Collaborative Software Development Environment



SoftwareForge.mil is the place to host public projects for software re-use/collaborative development

-  **Introduction to Forge.mil Webinar Replay: Slides Available**
For those that couldn't make it to the Introduction to Forge.mil Webinar, the slides for the presentation are now available. You can also watch a [replay](#) of the presentation as well.
-  **SoftwareForge Community Discussions**
Do you have a great idea for how to improve SoftwareForge.mil (new project category, best practice around development, etc.)? If so, please contribute to the SoftwareForge project [discussion area](#). This is your opportunity to help drive the development of this emerging community!
-  **Training for SoftwareForge Tools Available**
Do you have questions on how to use the SoftwareForge tool suite? Answers are available in the [site-wide training project](#). Computer-based "On Demand" training is available, and coming soon - instructor-led training. Please check back often for updates!

IEEE Std 802.3af™-2003
(Amendment to IEEE Std 802.3™-2000, including IEEE Std 802.3af™-2003)

802.3af™


IEEE Standard for Information technology—
Telecommunications and information exchange between systems—
Local and metropolitan area networks—
Specific requirements

Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications

Amendment: Data Terminal Equipment (DTE) Power via Media Dependent Interface (MDI)

IEEE Computer Society

Sponsored by the
LAN/MAN Standards Committee



Published by
The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5900, USA

14 June 2003

Price: \$100.00
PDF: \$50.00



Keys to Making SRP Technology Affordable, Effective (3 of 3)



- **Development Team Diversity**
 - SRP related strengths
 - Vested interests and expertise
 - Avoid the “winner-take-all” approach, or at least encourage large primes to diversify their team with subject-matter expert experts
 - Enabled diversified business models for application development
- **Government commitment helps to encourage success**
 - Adequately fund the project
 - Commit to building up infrastructure/ICDs, and other sharing mechanisms
 - Support conferences and other ways to share information



**Large
Aerospace
Company**



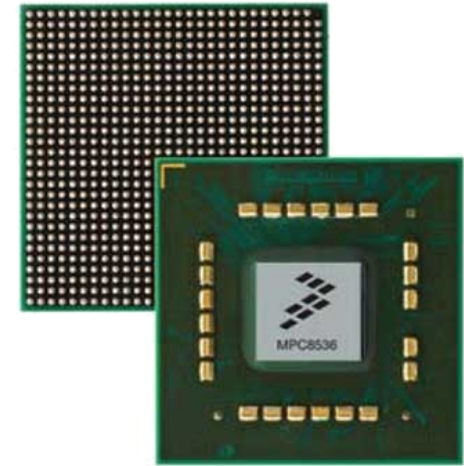
University

Small Business



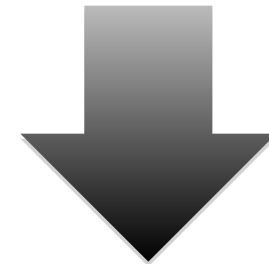
Provide Innovation Opportunities to Application Developers

- **SRP system provides multiple means for application developers to implement their area of their expertise**
- **Porting is straightforward**
 - **PowerPC compiler is open source (GNU)**
 - **FPGA development**
 - **MATLAB code**
 - **FPGA router tool**
 - **SRP “HAL” isolates FPGA interfaces from FPGA signal processing elements**
 - **Operating System**
 - **Linux for as long as possible for as many components as possible**
 - **Move to limited use of a proprietary OS if required due to security certification requirement**
 - **In either case, maintain POSIX compatibility**



Development Environment

- **Developers will be supplied government owned software development kit (SDK)**
- **SDK provide a software development environment at multiple levels of fidelity**
 - **Standalone [80%]**
 - Use a laptop or PC to develop software
 - Completely build and simulate on the laptop
 - **Developers kit / SRP Processor board [90%]**
 - **PowerPC Target**
 - **FPGA resources on SRP Processor board**
 - **SRP Transceiver card [100%]**
 - **PowerPC target**
 - **FPGA resources**
 - **RF resources**



Summary

- **Long acquisition timelines are a fact of life**
- **Building non-proprietary, government owned software and hardware designs allows a framework for enabling new disruptive acquisition models for new capability**
- **This requires the government to increase their involvement and go beyond being an acquisition agent but an agent of the program's success**
- **Tightly Coupled Government and Industry Team is Essential for Success**

USMC / NRL Software Reprogrammable Payload (SRP) Program Enables Rapid Response to Changing Needs While Ensuring Affordability - by Solving the Problem Once